UPLB-ORDeR: A Web Application for UPLB's Online Request of Documents and e-Signature Request System with Authentication and Verification using Smart Contract on Ethereum Blockchain

Shane Angela C. Palicpic and Danilo J. Mercado

***Abstract*— Ensuring the authenticity of a digital educational document is a must and responsibility of educational institution. UPLB-ORDeR is a system that guarantee the legitimacy of digital documents issued to UPLB students and not forged signatures of faculty members to documents. IPFS generates the document hash and uses Smart Contract to save immutable information of documents to Ethereum Blockchain. It also makes the verification process easier. System Usability Scale (SUS) is used to evaluate UPLB-ORDeR and based on mean SUS score, the application receives a Grade A or Excellent Usability.**

***Index Terms*— Blockchain, Ethereum, Smart Contract, Security, Authentication, Verification, Educational Document, IPFS, Infura**

I. Introduction

A. Background of the Study

The current COVID-19 pandemic, which began in China and practically infected almost every country in the world, is one of the most recent public health emergencies of global concern [1]. One of the most affected by this pandemic is the educational sector. The implementation of online learning platforms was promoted by responses like community quarantine in many nations, encouraging students and teachers to study and work from home [2]. In light of these circumstances, leaders in education decided to adopt a new normal in education.

Asking for educational records is included in the new norm in the educational system. Requesting papers must change from physical requests to online requests since students and officials work remotely. The legitimacy of the documents will be a worry because they are also converted to digital form.

The most important documents that institutions provide to their students are educational certificates. Since the issuance procedure is not verifiable, fake certificates can be easily created. It can be challenging to recognize fake certificates from real ones without a proper verification process [3].

Using blockchain technology is one solution to authenticate and verify digital documents. Satoshi Nakamoto proposed a well-known digital currency called Bitcoin whose foundation is Blockchain Technology. A blockchain is a revolutionary technology that is constantly evolving and being used in a variety of fields [4]. Originally used mainly for financial transactions, Blockchain is currently being suggested or implemented in every industry that requires safe and immutable record-keeping. It is a collection of blocks that have been formed using cryptographic hash functions and connected to one another in the form of a safe chain using cryptographic techniques [5]. This particular type of distributed ledger technology uses a peer-to-peer(P2P) network. A new block is verified using a consensus algorithm, once verified and put on the Blockchain, these blocks cannot be altered [3].

Blockchain technology is being deployed in a wide range of applications and is being suggested in many more due to the following characteristics:

a) Immutability: One of the most crucial aspects of blockchain is its ability to maintain its current state as a reliable and unalterable network;
b) Decentralized: The network doesn't have a single governing entity or administrator in the system;
c) Improved Security: Without a centralized authority, anyone can't just change network features to their advantage. The device is further protected via encryption;
d) Distributed Ledgers: Information about a transaction and its participants is in a public ledger. It's completely open so there is nowhere to hide;
e) Consensus: A consensus is a group of nodes that take part in the network's decision-making process. In this scenario, the nodes will accept with ease and without much difficulty [3].

Rouhani and Deters [6] stated that to manage the blockchain network's computing system infrastructure, a worldwide, open, and distributed network called Ethereum was developed. It is an open-source platform with a variety of features including smart contracts, ether, etc.

When a specific event occurs in any block, a smart contract will run, which is a piece of programming code posted on the Ethereum network. Across the blockchain network, smart contracts are self-executable, distributed, and shared [7].

Presented to the Faculty of the Institute of Computer Science, University of the Philippines Los Baños in partial fulfillment of the requirements for the Degree of Bachelor of Science in Computer Science

Smart contracts are computer programs that can operate independently on a blockchain under specific circumstances. A smart contract can be created from any protocol, flow, or programmable process. Whether or not the conditions mentioned in the contracts are met, the relevant procedures are automatically conducted, and the contract is completed. Smart contracts can be used to store data on the blockchain depending on the requirements, and it would be very advantageous to allow integration with current systems applied in the field of education [8].

University of the Philippines Los Baños (UPLB) is one of the institutions impacted by the transition to the new normal. From requesting documents physically by students to their respective colleges, it is now done online where instructions are posted on each college's Facebook page. Given that the university is one of the most prominent universities in the country, it is crucial to ensure the legitimacy of each document. With the help of smart contracts on the Ethereum blockchain, an application can be developed to authenticate and verify documents easily. Aside from that, the request for documents can be easily made by using only one web application specifically made for the use of the University.

B. Statement of the Problem

Before the pandemic, UPLB students request educational records from their respective College Secretaries on campus. With the new normal, UPLB College of Arts and Sciences (CAS) - Office of the College Secretary (OCS) released a guide on their Facebook about how to request documents online. One of the most important documents as a UPLB student is Form 5 which serves as the confirmation of enrollment and True Copy of Grades (TCG) mostly for scholarship purposes.

In order to request TCG, the student needs to see the instructions on the CAS-OCS Facebook page. Next, download the CAS-OCS Form 007.docx which can be found on the Google Drive folder given in the instructions. Once downloaded, use Microsoft Office Word or any software that can open and edit Docx files. Fill out all the necessary information on the form such as Date of Request, Complete Name, Student Number, Degree Program, UP Email Address, Mobile Number, and Student Classification if Freshman, Sophomore Junior, and Senior. The form also asks to choose the reason for requesting if it's for Extension, Job Application/OJT, Medical/Law School, Reinstatement, Scholarship, Transfer/Shift, Readmission, Graduate Study, Student Org Recognition, or Others. After that, the students can tick what kind of documents are being requested. If it is a True Copy of Grades then what semesters should be included as well as the Academic Year else the student can also request for certification of Bonafide Student, Completion of Academic Requirement, Completion of PE (PEPE), General Weighted Average (GWA), Good Moral Character signed by the College Secretary, Good Moral Character signed by the Dean, Graduation, Non-contract, Prospective candidate for graduation, Rank with (For graduated students only) GWA or GPA, Scholastic standing, Units earned, Units required in the curriculum, Year level/Classification or Others. Once done, fill out again another google form to upload the docx file and officially submit the request to CAS - OCS. After a few days, CAS-OCS will send an email containing the details on how to pay the requested documents through the landbank. When the payment is confirmed, then CAS-OCS will email the documents after another few days.

This method uses Facebook for instructions, Microsoft Office Word or other application that edits docx file to fill out personal information and document request, Google Forms to send the request, and Gmail to receive the documents. The UPLB students will repeatedly do all the processes if they will request documents. The documents can be easily modified using different document manipulating tools. Forged documents can also be created since there are no ways to check the verification of documents except by asking it directly to the CAS-OCS. It is a hassle for institutions who want to check the authenticity of the documents passed to them by UPLB students if they will contact the CAS-OCS directly just to ask for confirmation.

Additionally, UPLB students confirm their registration by filling out google forms issued by their respective college secretaries. This is done in order to get Form 5 which serves as the proof of their registration for the current semester. Form 5 contains courses enrolled for the semester and some personal information, it is always required to pass in any scholarship offices to get the student's stipend. As mentioned above, Form 5 can be easily modified or forged by students since there is no application to confirm its authenticity.

Signature forgery can also be easily made and it cannot be easily verified unless contacting the owner of the signature.

C. Objectives of the Study

This study aims to create a web-based application where UPLB students can request documents and signatures online. This application includes authentication and verification of the documents and signature using the smart contract of Ethereum Blockchain.

Specifically, the objective of this application is to:

1) Simplify the process for UPLB students to request documents online and for OCS staff to issue documents.
2) Simplify the process for UPLB students to request e-Signature to documents online and for Faculty members of UPLB to sign documents.
3) Make sure the authenticity of documents issued by the Office of College Secretary and signature of any Faculty members by using IPFS to generate document hash and Smart Contract to store immutable document data to Ethereum Blockchain.
4) Easier verification process to prove document's authenticity by just uploading a document.

D. Significance of the Study

UPLB-ORDeR can assist UPLB students in quickly requesting documents from their respective colleges. With this application, students can request documents without having to go to Facebook to know the instructions, view and edit the document request form with a document software, officially submit the request through google Forms and receive a copy

of the document requested through email. This reduces the application needed to open by just requesting one document. From multiple programs to a single web application where all the necessary steps to successfully request a document are present.

By logging in with UP Mail, each student has an account where they can save their personal information which is an advantage to not repeatedly filling out information for every document requested. By just choosing the document being requested, filling out the necessary fields on the website, and reviewing if the personal information is correct, the student is done with the request. The student can view the status of the request if it has been sent, in process, or done. All the documents previously requested are saved on the account of the students.

The website will be used by every UPLB student, thus every request is filtered per college. For instance, only the CAS-OCS account administrator or college's document issuer can see a CAS student's request. Knowing which requests still need to be processed is simple because the designated official or issuer can check the list of requests. Each document that is processed is authenticated using the smart contract of Ethereum Blockchain.

Institutions must ensure the authenticity of documents submitted to them. Contacting the other institution that issued the document is the traditional way to determine if it is authentic. Waiting for the other institution's confirmation or reply will make the processing time-consuming. Utilizing the website's verification feature, other institutions can quickly and easily verify the legitimacy of each document handled on the platform. No need to get in touch with the document's issuer.

To avoid the issue of forging signatures and make sure that the faculty member really signs a document, signature request is also a feature of the UPLB-ORDeR. A student can request a signature to a faculty member and attach the document to the request.

Efficiency is valued more these days. UPLB-ORDeR is one platform that will help to transform the document request system in UPLB. Students can use only one website to request documents. Both students and the issuer can track the status of the documents. Each document is authenticated and can be easily verified. As we move toward the new norm, security is crucial for all digital transactions, especially those involving student records.

II. Review of Related Literature

All aspects of people's lives have recently been impacted by the digital age where online access to information is convenient. The internet, though, is not about value. The ease of access provided by the internet opens a loophole in how crime is committed. One example is the falsification of academic credentials [9].

Academic credentials are highly valued since they reveal the human capital of the holders. The skills, competencies, knowledge, and aptitudes acquired via education are referred to as human capital. Fake academic certificates can be obtained from five main sources. These include "Degree Mills," where fictitious credentials are created and sold to customers; "Fabricated Documents," which portray a fictitious degree or institution; "Modified Documents," which are modifications made to genuine documents, such as adjustments to enrollment or graduation dates, grades, course content, dates of birth, specialization, etc.; "In-House Produced," which are false documents created by staff members of legitimate institutions and printed on real paper; and "Translations" or documents that have been altered to meet the needs of a receiving country. When communicating with the issuing institution, verification typically takes days or weeks. Even though document verification is time-consuming, it is necessary to prove a document's legitimacy. The most latest technology to make the authentication and verification of academic records easier is blockchain [10].

A. Blockchain in Education

The University of Nicosia (UNIC) uses the Bitcoin blockchain for a variety of purposes, including taking bitcoin as payment for any degree program's tuition and issuing academic credentials on the blockchain [11]. The University of Nicosia's initiative to create educational diplomas on the blockchain aims to combat financial fraud among international students. The major objective is to solve the issues caused by changing the cohort sizes of students. Since 2017, UNIC has started using the blockchain to issue all diplomas, and it offers users software tools to verify the legitimacy of the certificate. UNIC is a part of the Blockcerts consortium and their user-facing systems adhere to the current open source standards. The SHA-256 hash technique is utilized when sharing certificates with another entity as a PDF file. Because a hash can be generated from a certificate using SHA-256, but not the other way around, this algorithm is widely used. Searching the index document's SHA-256 for the certificate preserves the authenticity of the certificate. The certificate is legitimate if the code matches. Despite these protections for the privacy, ownership, and integrity of the certificate, work still needs to be done to make the hash publicly verifiable. This is one condition for letting employers inspect the certificate. Furthermore, it's possible that the recipient will be unable to permit a prospective employer to use the hash to verify the certificate [12].

To give recipients more power over the certifications they acquire, the MIT Media Lab issues digital certificates to groups of students using Blockcerts. The recipients of this initiative may not be required to store, confirm, and validate credentials through a third-party intermediary. The MIT certification architecture relies on the process of the issuer signing a digital certificate and storing its hash within the blockchain transaction. The recipient is given ownership of the transaction's output. The ownership issue arose in this initiative because MIT had to roll out certificates based on user-generated key pairs for their graduation and workshop participation certificates. In addition, a high level of trust is necessary for this system. The hashing method is employed when an employer needs to confirm the legitimacy of a certificate and must disclose both the certificate and the hash of

the certificate stored on the blockchain. The theme of privacy is not entirely satisfied in this instance [13].

Another tool for verifying digital credentials based on blockchain is SmartCert. A blockchain-based system called SmartCert was created to verify the legitimacy of academic credentials and solve the issue of false certificates. To guarantee transparency in the case of hiring, SmartCert uses the digital signing of educational certificates. In order to verify the certificate, the student will provide the hash to the potential employer. A legitimate user may find it challenging to access data that is protected by a hash or a digitally signed certificate since the computer that is viewing the data may be hacked by a hacker. Another problem with this application is that basic security precautions must be taken to protect against threats because cryptography does not guarantee data protection. At the same time, SmartCert's cryptographically secured certificates prevent easy certificate fraud [14].

Another blockchain-based tool for verifying academic credentials is RecordsKeeper. Educational institutions can issue certificates using RecordsKeeper and give the recipient a receipt they can show to a third party as proof the certificate is genuine. The third party will utilize the receipt they received from the student to check the legitimacy of the certificate in the RecordKeeper ledger. The parties interested in viewing the certificate on the Record Keeper blockchain must have ownership rights, but there aren't many other issues with this approach. This essentially equates to giving the third-party ownership, which could result in tampering. To assure the security of the certificate, this might function effectively on a private blockchain [15].

B. Digital Authentication and Verification

The end of a paper-based certificate system may come sooner due to blockchain technology. Since the legitimacy of digital certificates can be checked against the blockchain, the blockchain enables institutions to issue permanently valid, immutable certificates. The value proposition of digital certificates is greatly increased by these benefits over existing methods, which is anticipated to drive the adoption of digital certification [14]. The requirement for educational institutions to verify credentials is eliminated by blockchain technology. Educational institutions won't have to devote resources to certificate verification because certificates created on the blockchain can be done so automatically [16].

Digital certificates stored securely on a blockchain have a number of advantages over regular digital certificates. Anyone with access to the blockchain can quickly and easily verify the certificate's authenticity using freely accessible open-source software. Consequently, the need for middlemen has been eliminated. As a result, the certificate can still be verified even after an entity has been dissolved or lost access to the issued record. On a blockchain, issued records and acquired certificates can be deleted if all copies on all computers hosting the software are destroyed. The user holds the hash, which establishes a link to the original document. The system permits the publication of the document signature but does not mandate the publication of the actual document. This system keeps the document's data private [10].

C. Ethereum Blockchain

Vitalik Buterin first proposed the concept of Ethereum in 2013, and it was operationalized in 2015. It is stated that Ethereum was raised to $18 Million in just 42 days in 2014, when Buterin was 19 years old. The programmable blockchain service Ethereum is open to user contributions [17]. Ethereum, like other cryptocurrencies, makes use of blockchain technology. The ability to create and execute "smart contracts" and "distributed autonomous applications-DApps" is the feature that distinguishes Ethereum from competing cryptocurrencies [18]. These applications and smart contracts are created using the programming language "Solidity," which is then compiled and converted to bytecode and sent to the Ethereum blockchain as a Smart Contract.

The Ethereum Virtual Machine (EVM) powers blockchain applications and smart contracts. Costs associated with smart contract operation on the blockchain and transaction approval include contract size in bytecode, volume of data transferred, and transaction fees. During the deployment and operation of the smart contract, these fees are displayed as Ether or Gas as a sub-unit [19]. The creation of distributed autonomous applications or smart contracts on actual blockchains has the potential to cause enormous financial losses due to coding errors. As a result, it would be prudent to send to the Ethereum blockchain after conducting local/personal testing first, testing against all potential outcomes in test blockchains like Ropsten, Kovan, and Rinkeby. These networks, known as the Testnet, allow applications to be tested in all dimensions with any number of users without incurring any fees [8].

Ether. Ethereum-based applications use the cryptocurrency ether. Cryptocurrency is a type of digital money used for trading in digital transactions. For any event, ether serves as a transaction charge. Ethereum thus offers programmers a platform from which to build any kind of decentralized application for the blockchain network [20].

Smart contracts. Smart contracts are computer programs that can operate independently on a blockchain under specific circumstances. A smart contract can be created from any protocol, flow, or programmable process. Whether or not the conditions mentioned in the contracts are met, the relevant procedures are automatically conducted, and the contract is executed. Smart contracts can be used to store data on the blockchain depending on the requirements, and it would be very advantageous to allow integration with current systems employed in the field of education [8].

Solidity. Solidity is a popular high-level programming language used to build smart contracts on various blockchain networks, such as Ethereum. Solidity is an object-oriented programming language. It draws inspiration from various well-known OOP-featured programming languages, such as C++, Python, and JavaScript. Solidity is intended to run smart contracts that have been deployed in the Ethereum Virtual Machine (EVM). With the help of the solidity language, business and computational logic are encoded in smart contracts. Solidity provides all the tools necessary to create smart contracts, making the process of doing so comparatively simple [20].

InterPlanetary File System. The Peer to Peer (P2P) network

protocol for data storage, distribution, and transfer is known as the InterPlanetary File System (IPFS). IPFS links all computers globally by using a content-based addressing mechanism to locate each file independently. With this, a user can both get content from any node that possesses the desired content and host any content for other network users. The IPFS system offers flexible file storage and distribution mechanism since a portion of the total data is held by certain user operators. By utilizing its distinctive content, any network user can host a data file or other piece of information [20].

III. Methodology

A. System Architecture

The UPLB-ORDeR's system architecture is shown in Figure 1.

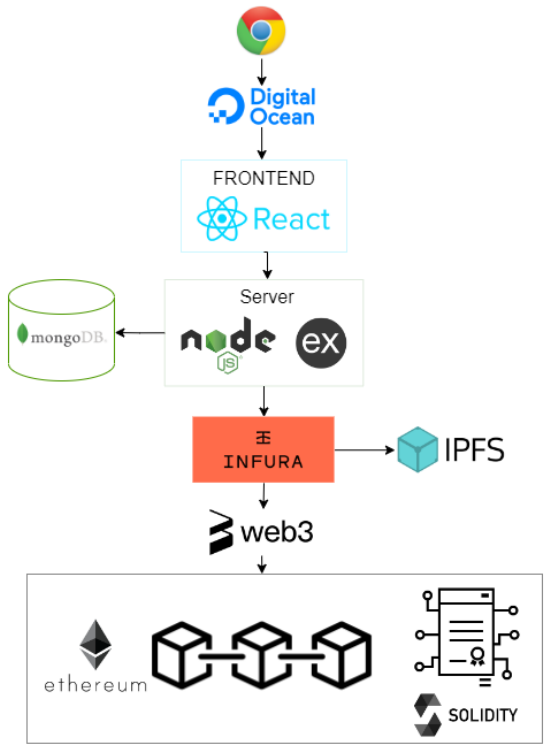

Fig. 1: Architecture Diagram of UPLB-ORDeR

UPLB-ORDeR is deployed on DigitalOcean Droplet Ubuntu 20.04 distribution with Basic shared CPU Premium AMD with NVMe SSD, 2GB RAM/1 AMD CPUs, 50GB NVMe SSDs and 2TB transfer.

The web application is developed using the following:

Frontend: ReactJS is a JavaScript library for building user interfaces.

Backend: ExpressJS is a straightforward and flexible Node.js web application framework that gives web applications a wide range of features.

Database: MongoDB is a document database used to create scalable and highly accessible web applications. The student records and list of documents requested are stored in this database.

Smart Contract: Solidity is a high-level, object-oriented language that can be used to implement smart contracts and interact with the blockchain.

The server calls the Infura which provides instant, scalable API access to the Web3 and IPFS networks. The following are some libraries used:

- **Web3**: a collection of libraries that allow you to interact with a local or remote Ethereum node using HTTP, IPC, or WebSocket. This will be used to deploy and interact with the smart contracts.
- **IPFS**: Documents requested and authenticated will be stored in IPFS so that it will generate a unique hash.
- The library @truffle/hdwallet-provider connects web3 to an Ethereum account to use the Goerli Testnet Balance in order for transactions to proceed.

B. Database Model

As shown in Figure 2, the database of UPLB-ORDeR only has 4 Entities: User, Document Request, Signature Request and Contract. A User can have zero to many Document Request or Signature Request. A Document Request may have zero to one Contract and requires one user. Signature Request requires one user and recipient but Contract is only optional.

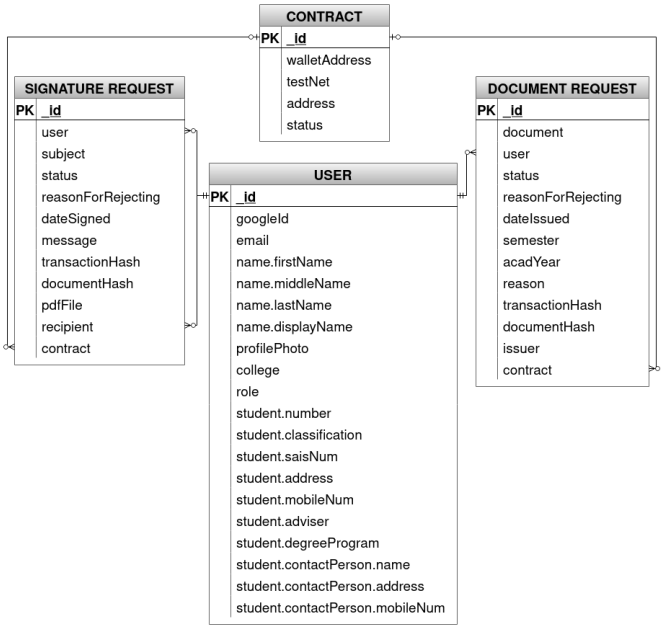

Fig. 2: Database Model of UPLB-ORDeR

C. Use Cases

The Application consists of 5 User Roles: Student, Faculty, OCS Staff, Admin and Guest User. In order to visualize how the system works per user role, Figure 3 contains the use case diagram.

As shown in Figure 3, all the users can verify documents on the website. Students can Request Document and Request Signature. OCS Staff can Issue Document. Faculty can Sign Document. And lastly, Admin can Manage Users, Manage Smart Contracts and View Documents authenticated on Blockchain.

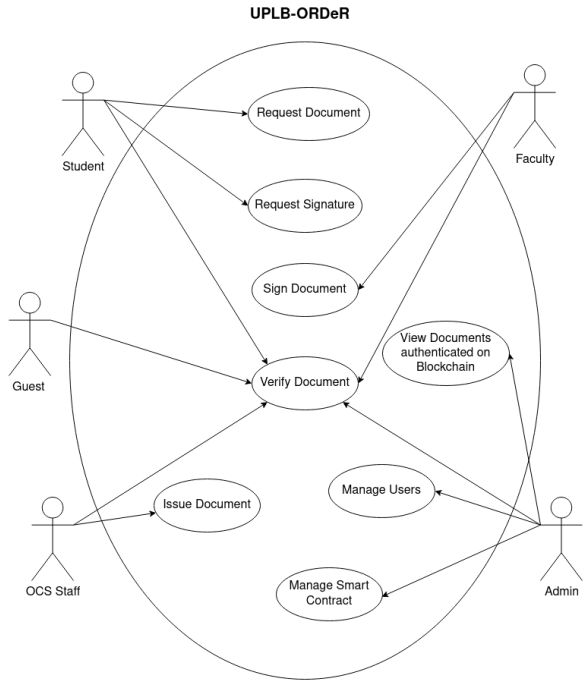

Fig. 3: Use Case Diagram of UPLB-ORDeR

D. Function Specification

These are the detailed function of each user.

1) **Student** - UPLB Undergraduate Student
   a) Login using Google
      i) The first name, middle name, last name, displayName and photo are retrieve from the google account.
      ii) Choose role (user type) on the page on first login.
   b) Document Request
      i) Shows all document request history of the user sorted by the latest first.
      ii) Document Request list can be filtered by status (Pending, Issued, Denied).
      iii) View all details of a specific request.
      iv) Request Document to respective Office College of Secretary
         A) All fields cannot be empty.
         B) Fields vary when the Document Name is "Form 5", "True Copy of Grades" and "Others:", Otherwise, the fields get basic Document Request Information.
   c) Signature Request
      i) Shows all signature request history of the user sorted by the latest first.
      ii) The signature Request list can be filtered by status (Pending, Signed, Denied).
      iii) View all details of a specific request.
      iv) Request document signature to any faculty listed in the database.
         A) Specify Recipient, Subject, Message and File to sign.
         B) Can request the same document to different Faculty.
   d) Verify Document
      i) Upload document to verify
2) **OCS Staff** - OCS Staff of any undergraduate colleges in UPLB who issues document to students
   a) Login using Google
      i) The first name, middle name, last name, displayName and photo are retrieve from the google account.
      ii) Choose role (user type) on the page on first login.
   b) Issue Document
      i) Shows all the document requests sent to the user's college sorted by latest first.
      ii) Document Request list can be filtered by status (Pending, Issued, Denied).
      iii) View all details of a specific request.
      iv) Authenticate and issue document.
      v) Reject a document request.
   c) Verify Document
      i) Upload document to verify
3) **Faculty** - UPLB Faculty Member/Teaching Staff/Lecturer
   a) Login using Google
      i) The first name, middle name, last name, displayName and photo are retrieve from the google account.
      ii) Choose role (user type) on the page on first login.
   b) Sign Document
      i) Shows all the signature requests sent to the user sorted by latest first.
      ii) Signature request list can be filtered by Status (Pending, Signed, Denied)
      iii) View all the details of a specific request.
      iv) Sign a document.
      v) Reject a signature request.
   c) Verify Document
      i) Upload document to verify
4) **Admin** - Manages the application or the developer of the app
   a) Login using Google (Admin account)
      i) The first name, middle name, last name, displayName and photo are retrieve from the google account.
   b) Manage Users
      i) Shows all Users sorted by College.
      ii) Users can be filtered according to their Role (Student, Faculty, OCS Staff, Admin).
      iii) Edit College and Role of the User.
      iv) Delete a User.
   c) Manage Smart Contracts
      i) Shows deployed Smart Contract.
      ii) Edit Smart Contract's status to inactive or active.
      iii) Contracts can be filtered according to their status (Active, Inactive).

iv) Deploy new Smart Contract if no contract is deployed or active and there are changes in Smart Contract functionality.

d) Retrieve Documents' Information saved on the Blockchain.

i) Shows all the document data saved on Ethereum Blockchain of an active deployed contract.

e) Verify Document

i) Upload document to verify

5) **Guest Users** - Users who can use the application without logging in for verification of documents.

a) Verify Document

i) Upload document to verify

E. Web3

The functionalities that calls Web3 api are Issue Document, Sign Document, Verify Document, Deploy Smart Contract and Show all document record on blockchain.

The process when Issuing or signing documents is the IPFS will generate the document hash, and save other information like student email, issuer email and signature emails on the ethereum blockchain network using the smart contract deployed.

Verifying document is also generating the document's hash from IPFS and checking if the document's hash matches the system's record of that document. The data will then be retrieved from the blockchain network using the document hash and smart contract.

For every access to the blockchain network, a gas fee is paid using the Ethereum Wallet account's Goerli testnet balance. Since it is a testnet, no actual money is paid to have balance on the account but rather the balance is mined using Goerli faucet available on the net.

F. User Evaluation

The application is intended for UPLB constituents' advantage. In order to test the UPLB-ORDeR, the developer will employ the user participation approach. The developer will look for willing UPLB students, Faculty Member/Teaching Staff/Lecturer, OCS Staff to test the app. On the other hand, non-UPLB constituents are also allowed to test as a Guest User. The testing of the web application will be done purely online.

Willing participants will have tasks given to them depending on their role. Participants need to access the website and explore its usability and user interface. A google form will be provided so the participants can test the website's usability, functionality, and overall judgments about the system.

System Usability Scale (SUS) and open ended questions are included in the form. The questionnaire's numeric scale is from 1 to 5, where 5 is the respondent's strong agreement with the statement and 1 is being strongly disagree. The respondents will be asked if UPLB-ORDeR is more advantageous than the usual system in UPLB and if they prefer it more. The participants will be asked open-ended questions in the final section of the Google Form to list the benefits and drawbacks of using UPLB-ORDeR for more application development and improvement.

IV. Results and Discussion

A. Web Development

The user interface of UPLB-ORDeR takes into account the University of the Philippines color and using Material UI in React, the following are the outcome of the user interface:

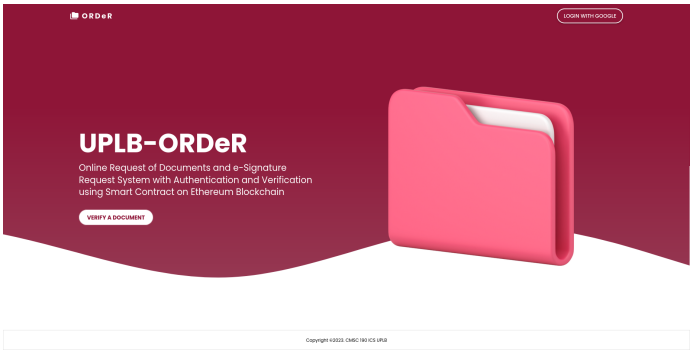

Fig. 4: Homepage of UPLB-ORDeR

The Figure 4 contains the title of the website, login button and verify document button.

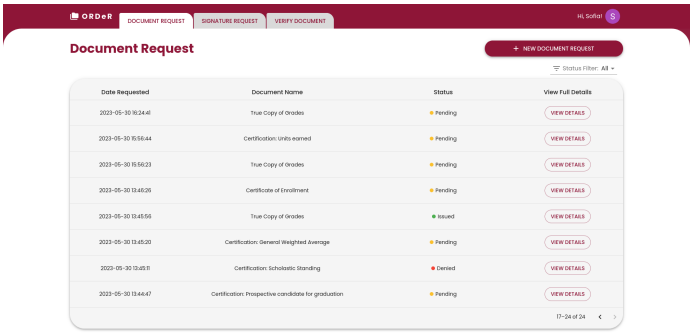

Fig. 5: Document Request Page of UPLB-ORDeR

The overall main page of Document Request and Signature Request are similar in terms of layout and design of Figure 5, the only difference are the labels and text. On the other hand, Issue Document and Sign Document page are also similar, only without the button on the upper-right part of the table.

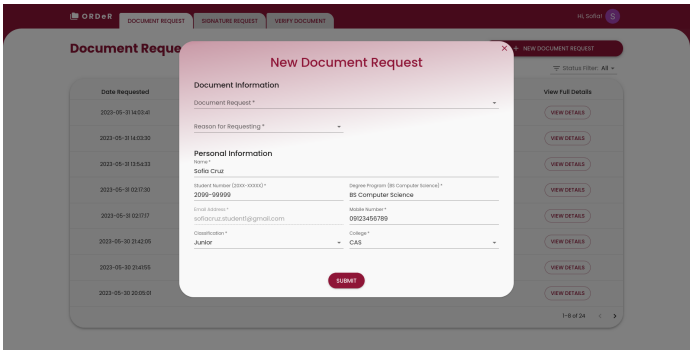

Fig. 6: New Document Request Modal of UPLB-ORDeR

The new document request modal on Figure 6 ask for Document Information and Personal Information.

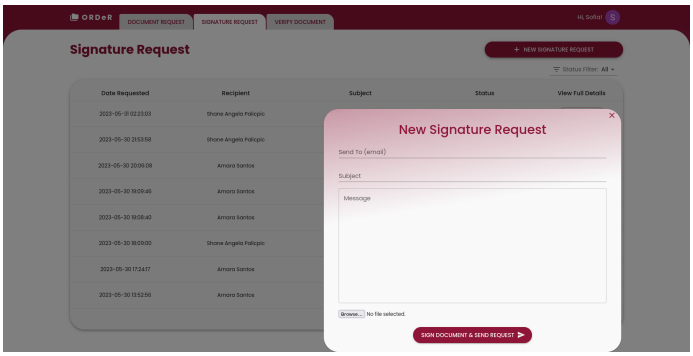

Fig. 7: Signature Request Modal of UPLB-ORDeR

Recipient email, subject, message and pdf file is required on the new signature request modal on Figure 7.

In order for OCS Staff to issue document, after clicking view details of specific pending request the modal is shown in Figure 8.

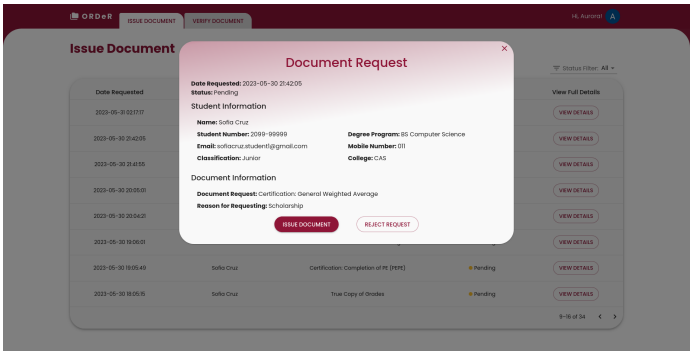

Fig. 8: Pending Document Request Modal of UPLB-ORDeR (OCS Staff View)

When Issue Document is clicked, the Figure 9 shows the upload document field become available.

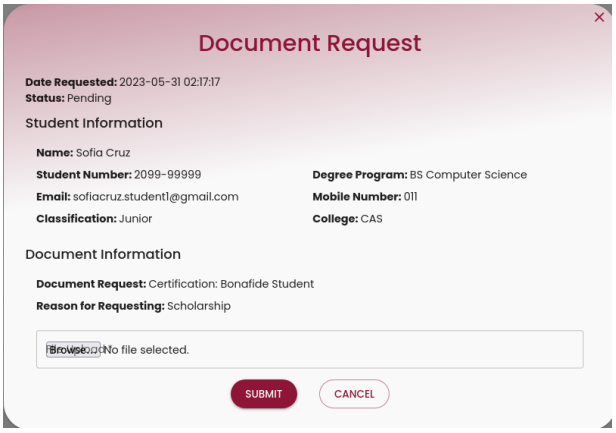

Fig. 9: Issue Document Modal of UPLB-ORDeR (OCS Staff View)

When Reject Document is clicked, text field for reason will be available as shown on Figure 10.

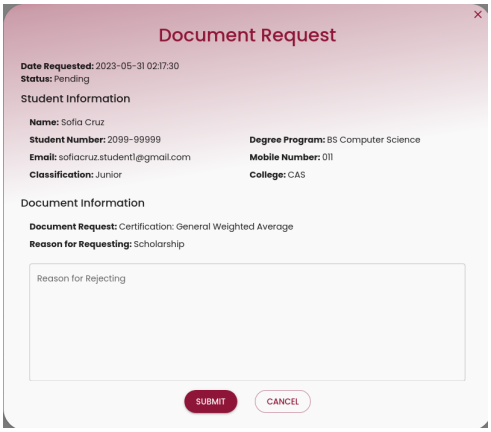

Fig. 10: Reject Document Request Modal of UPLB-ORDeR (OCS Staff View)

The details of Document Request such as Student Information and Document Information with Issued status is shown in Figure 11.

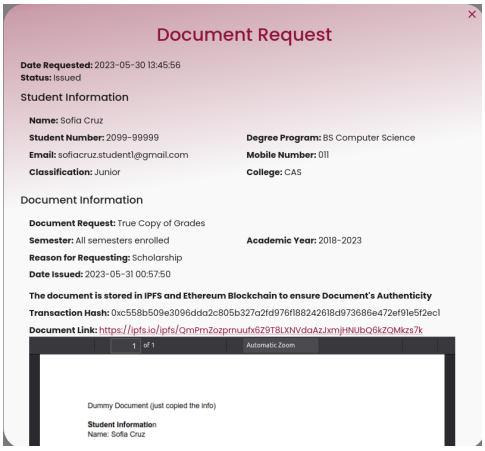

Fig. 11: Issued Document Information Modal of UPLB-ORDeR (OCS Staff and Student View)

Figure 12 shows the Document Request information when status is denied.

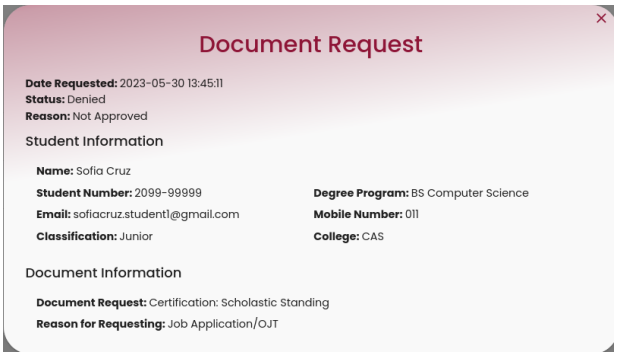

Fig. 12: Rejected Document Request Information Modal of UPLB-ORDeR (OCS Staff and Student View)

Faculty can sign a document by clicking View Details on specific request. The student's information who request

signature and signature request information are shown as well as the document. In Figure 13, it shows that the Faculty can sign or reject a signature request.

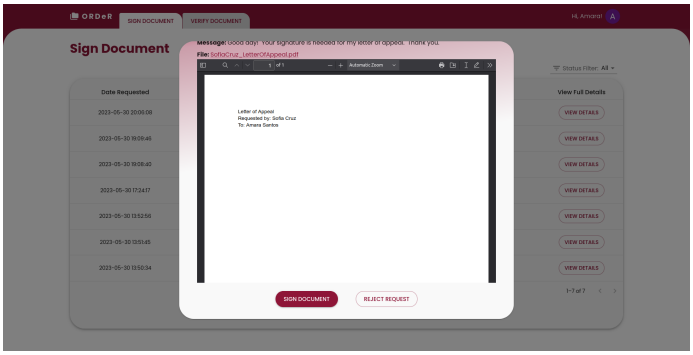

Fig. 13: Pending Signature Request Modal of UPLB-ORDeR (Faculty View)

When document signing is successful, the modal on Figure 14 shows all the details about the request, the view is the same in student and faculty.

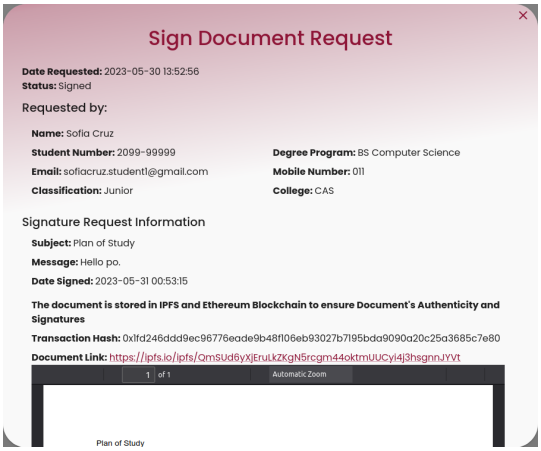

Fig. 14: Signed Document Modal of UPLB-ORDeR (Student and Faculty View)

Admin Page's first tab is Manage User in Figure 15, it shows all the users and admin can edit the role and college and delete a user.

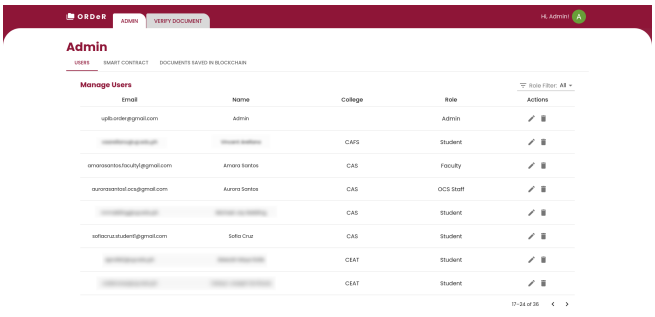

Fig. 15: Manage User of UPLB-ORDeR

In Figure 16, the button to deploy new contract is disabled because there is an active contract. The admin can edit the status of a deployed smart contract.

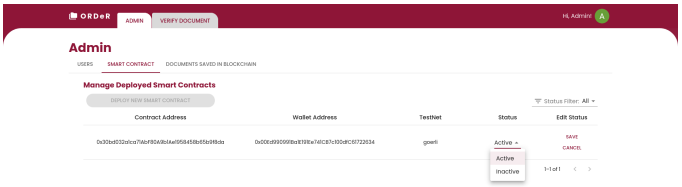

Fig. 16: Manage Smart Contract of UPLB-ORDeR

Once the button to show documents is clicked on Figure 17, it will show all the document records saved by the active smart contract on Ethereum blockchain. It is not automatically show on its tab because it will always spend gas fee.

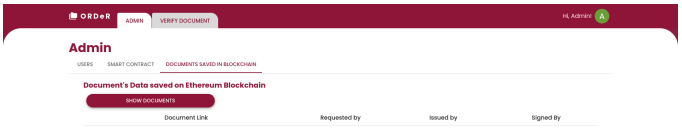

Fig. 17: Retrieve Documents' Information saved on the Blockchain Page of UPLB-ORDeR

Figure 18 shows the Verify Document page, document's authenticity can be verified by just uploading it.

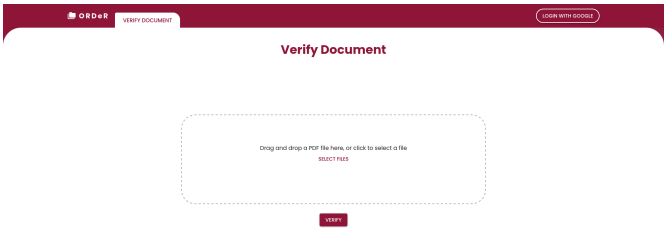

Fig. 18: Verify Document Page of UPLB-ORDeR

If the document is authentic, the details of the document can be seen in Figure 19

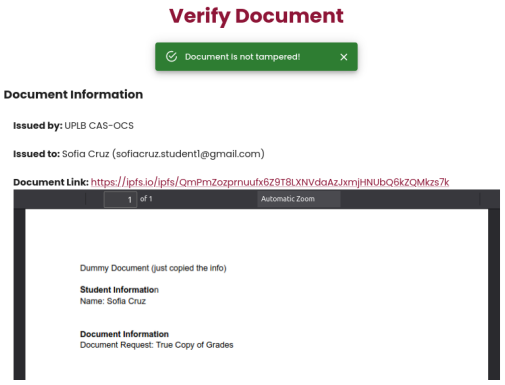

Fig. 19: Authentic Document in Verify Document Page of UPLB-ORDeR

B. Evaluation Results

A total of 32 users take part in the evaluation that consists of 28 Students, 3 Faculty and 1 OCS Staff all within UPLB. After testing based on procedure given to them and exploring the website on their own, the users evaluate the application using System Usability Scale (SUS) and some open ended questions.

Table I shows the SUS frequency distribution of the evaluation results and the table header consists of Statement and labels SD for Strongly Disagree, D for Disagree, N for Neutral, A for Agree, SA for Strongly Agree and T for the Total frequency of each row.

Statements	SD	D	N	A	SA	T
I think that I would like to use UPLB-ORDeR frequently.	1	0	0	11	20	32
I found UPLB-ORDeR unnecessarily complex.	17	10	2	1	2	32
I thought UPLB-ORDeR was easy to use.	0	1	2	6	23	32
I think that I would need the support of a technical person to be able to use UPLB-ORDeR.	14	11	3	4	0	32
I found the various functions in UPLB-ORDeR were well integrated.	0	0	1	12	19	32
I thought there was too much inconsistency in UPLB-ORDeR.	18	13	1	0	0	32
I would imagine that most people would learn to use UPLB-ORDeR very quickly.	0	0	1	8	23	32
I found UPLB-ORDeR very cumbersome to use.	20	11	1	0	0	32
I felt very confident using UPLB-ORDeR.	0	1	2	10	19	32
I needed to learn a lot of things before I could get going with UPLB-ORDeR.	14	13	3	2	0	32

TABLE I: Frequency distribution of the results in System Usability Scale (SUS)

According to the table, odd numbered statements are mostly leaning to Strongly Agree while even numbered statements are leaning towards Strongly Disagree.

Score	Grade	Score	Grade	Score	Grade	Score	Grade
100	A	95	A	87.5	A	80	B
100	A	92.5	A	87.5	A	80	B
100	A	90	A	87.5	A	77.5	B
100	A	90	A	87.5	A	70	B
100	A	90	A	85	A	70	B
97.5	A	90	A	85	A	67.5	D
97.5	A	90	A	82.5	A	60	D
97.5	A	87.5	A	82.5	A	52.5	D

TABLE II: Calculated SUS Scores

Mean Score: 86.25

Score	Grade	Interpretation
> 80.3	A	Excellent
68 – 80.3	B	Good
68	C	Okay
51 – 68	D	Poor
< 51	F	Awful

TABLE III: SUS Score Interpretation Brackets

Table II shows the calculated SUS scores of 32 participants from highest to lowest. The scores are graded based on SUS Score Interpretation Brackets on Table III. The resulting mean score of all calculated SUS score is 86.25 which is is within the grade A and Excellent interpretation.

Aside from SUS, the users were also asked questions about the application. The usual comment about the overall design and user interface are good in terms of color palette and readability of things, pleasing in the eyes, and user friendliness. Being organized, and simplicity of design and layout are what the users like the most about the website. On the other hand, the lack of information within the website is what they like the least.

Based on faculty member testers, one observe that the document take some time to load after signing it and the disadvantage is no notification when a student request a signature. However, it is beneficial to them in terms of easy signing of documents that can be verified.

OCS Staff tester states that it can be very helpful to track the document request if already issued it or not but no idea on the choices of students about the name of documents to request and list of reasons for requesting.

According to the testers, the advantage of UPLB-ORDeR compared to the current methods used in the document request and signature request within the University is it being straightforward, faster to request, does not require much details when requesting, transaction status is present, easy to use and use of blockchain for authentication. However, some disadvantages are expenses of transactions, it might intimidate some non-techy users, Faculty and OCS Staff cannot authenticate documents at night due to network congestion at blockchain every night.

All users would like to recommend UPLB-ORDeR to UPLB.

In general, the overall thoughts of users about the application are: it will be a huge help to students in terms of easier requests; good user interface; and users are hoping that UPLB will implement it in the future.

V. Conclusion and Recommendation

User's evaluation resulting in grade A Excellent interpretation of SUS mean score proves UPLB-ORDeR's usability and how beneficial the application is to students, faculty and OCS Staff of UPLB. However, it includes limitations that can hinder production when handed over to UPLB.

A. Limitation

As mentioned, every access to Ethereum blockchain requires gas fees that are deducted from the wallet balance of the Ethereum account used. Additionally, the system uses Infura API which limits the requests to Web3 by 100,000 per day and 1GB of total documents saved to IPFS with their free tier package.

On Table IV, using the link goerli.etherscan.io and the transaction hash is entered, all the transaction data can be seen such as the transaction fee. The following are successfully Issued and Signed Documents, it shows that the price per

transaction varies on different days and price ranges from 55 pesos to 285 pesos.

Date	Action	Transaction Hash	Transaction Fee	Price in Pesos 1 ETH = 105,183.74 PHP (June 5, 2023)
June 2, 2023	Issue Document	0xc558b509e3096dda2c805b327a2fd976f188242618d973686e472ef91e5f2ec1	0.00121256 ETH	127.54 PHP
May 31, 2023	Issue Document	0xb5d4474907f66e6fce2dd4f1d3a7ad81539b01e999a04868f403db13d631dd38	0.00220453 ETH	231.88 PHP
May 31, 2023	Issue Document	0xcb90b9ecdc15211bec3c68465215ec3f9114893c4b0b75ae635a96993f870489	0.00220453 ETH	231.88 PHP
June 2, 2023	Signed Document	0x00c0d893f258078a3d39fd08e4d47310feed62384af5c3c5aab9981bd4e93b06	0.00052974 ETH	55.72 PHP
May 31, 2023	Signed Document	0x1fd246ddd9ec96776eade9b48f106eb93027b7195bda9090a20c25a3685c7e80	0.00271012 ETH	285.06 PHP
May 31, 2023	Signed Document	0xf112181bfd15448bd8f9840f06e4b7da9c2deb7315d76ecb0fce517813d8a3e7	0.00271012 ETH	285.06 PHP

TABLE IV: Transaction Record Information of Each Document

This means that the system requires a lot of money for the expenses of using blockchain technology given that the Ethereum price right now is high.

B. Recommendation

There is always room for improvement which the users specifically recommended. Some adjustments to User Interface can be made like centering the modal for signature requests, the application is responsive but also adjust the design on mobile size, and add more information on the page. Some bugs are evident such as accepting non-PDF files and accepting 1000 words on fields.

Features that may be good to add to the application are also mentioned such as two-factor authentication, verify role for first time users, dashboard page, specify if the document is rush when requesting, User Profile, more field validation, notification system such as notifying via email, payment method, information tab, Contact page with the list of faculty members, an inbox like feature where a student can place inquiry, can report a bug, deadline of signing a document (suggested by faculty member tester) and information about the students' choices of documents name and reason for requesting (suggested by OCS Staff).

Given the limitations mentioned, the idea of authenticating and easy verification of documents using web3 is pricey so it is better to look for more options that do not require large amount of money.

Overall, UPLB-ORDeR's purpose to secure educational documents' authenticity with the use of smart contract in Ethereum blockchain while simplifying the process of document request and signature within UPLB was done accordingly.

REFERENCES

[1] Y.-R. Guo, Q.-D. Cao, Z.-S. Hong, Y.-Y. Tan, S.-D. Chen, H.-J. Jin, K.-S. Tan, D.-Y. Wang, and Y. Yan, "The origin, transmission and clinical therapies on coronavirus disease 2019 (covid-19) outbreak – an update on the status," *Military Medical Research*, vol. 7, 2020. [Online]. Available: https://doi.org/10.1186/s40779-020-00240-0

[2] J. Crawford, K. Butler-Henderson, J. Rudolph, B. Malkawi, M. Glowatz, R. Burton, P. A. Magni, and S. Lam, "Covid-19: 20 countries' higher education intra-period digital pedagogy responses," *Applied Learning and Teaching*, April 2020. [Online]. Available: https://journals.sfu.ca/jalt/index.php/jalt/article/view/191

[3] S. Alam, "A blockchain-based framework for secure educational credentials," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, pp. 5157–5167, 2021.

[4] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized Business Review*, 2008.

[5] M. Aamir, R. Qureshi, F. A. Khan, and M. Huzaifa, "Blockchain based academic records verification in smart cities," *Wireless Personal Communications*, vol. 113, pp. 1397–1406, 2020.

[6] S. Rouhani and R. Deters, "Blockchain based access control systems: State of the art and challenges," *IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 423–428, October 2019.

[7] F. Buccafurri, G. Lax, L. Musarella, and A. Russo, "Ethereum transactions and smart contracts among secure identities," *DLT@ ITASEC*, pp. 5–16, 2019.

[8] E. KARATAŞ, "Developing ethereum blockchain-based document verification smart contract for moodle learning management system," *Bilişim Teknolojileri Dergisi*, vol. 11, pp. 399–406, 2018.

[9] S. K. UntungRahardja and Q. EkaPurnamaHarahap, "Authenticity of a diploma using the blockchain approach," *Int. J.*, vol. 9, 2020.

[10] O. Ghazali and O. S. Saleh, "A graduation certificate verification model via utilization of the blockchain technology," *Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 10, pp. 29–34, 2018.

[11] Y. Perez. (2015) The global universities embracing cryptocurrency. [Online]. Available: https://faculty.fuqua.duke.edu/∼charvey/Media/2015/CD_September_14_2015.pdf

[12] F. Bond, F. Amati, and G. Blousson, "Blockchain, academic verification use case," *Buenos Aires*, 2015.

[13] O. S. Saleh, O. Ghazali, and M. E. Rana, "Blockchain based framework for educational certificates verification," *critical reviews*, vol. 7, pp. 79–84, 2020.

[14] T. Kanan, A. T. Obaidat, and M. Al-Lahham, "Smartcert blockchain imperative for educational certificates," *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, pp. 629–633, April 2019.

[15] (2021, June) Verify academic certifications. RecordsKeeper. [Online]. Available: https://www.recordskeeper.com/verify-academic-certifications/

[16] E. C. Garwe, "Qualification, award and recognition fraud in higher education in zimbabwe," *studies in education*, vol. 5, pp. 119–135, 2015.

[17] J. M. Newman, "Innovation policy for cloud-computing contracts," *Research Handbook on Digital Transformations*, pp. 380–399, 2016.

[18] V. Gatteschi, F. Lamberti, C. Demartini, C. Pranteda, and V. Santamaría, "Blockchain and smart contracts for insurance: Is the technology mature enough?" *Future internet*, vol. 10, p. 20, 2018.

[19] V. Buterin, "A next-generation smart contract and decentralized application platform," *white paper*, vol. 3, pp. 2–1, 2014.

[20] I. T. Imam, Y. Arafat, K. S. Alam, and S. A. Shahriyar, "Doc-block: A blockchain based authentication system for digital documents," *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pp. 1262–1267, February 2021.